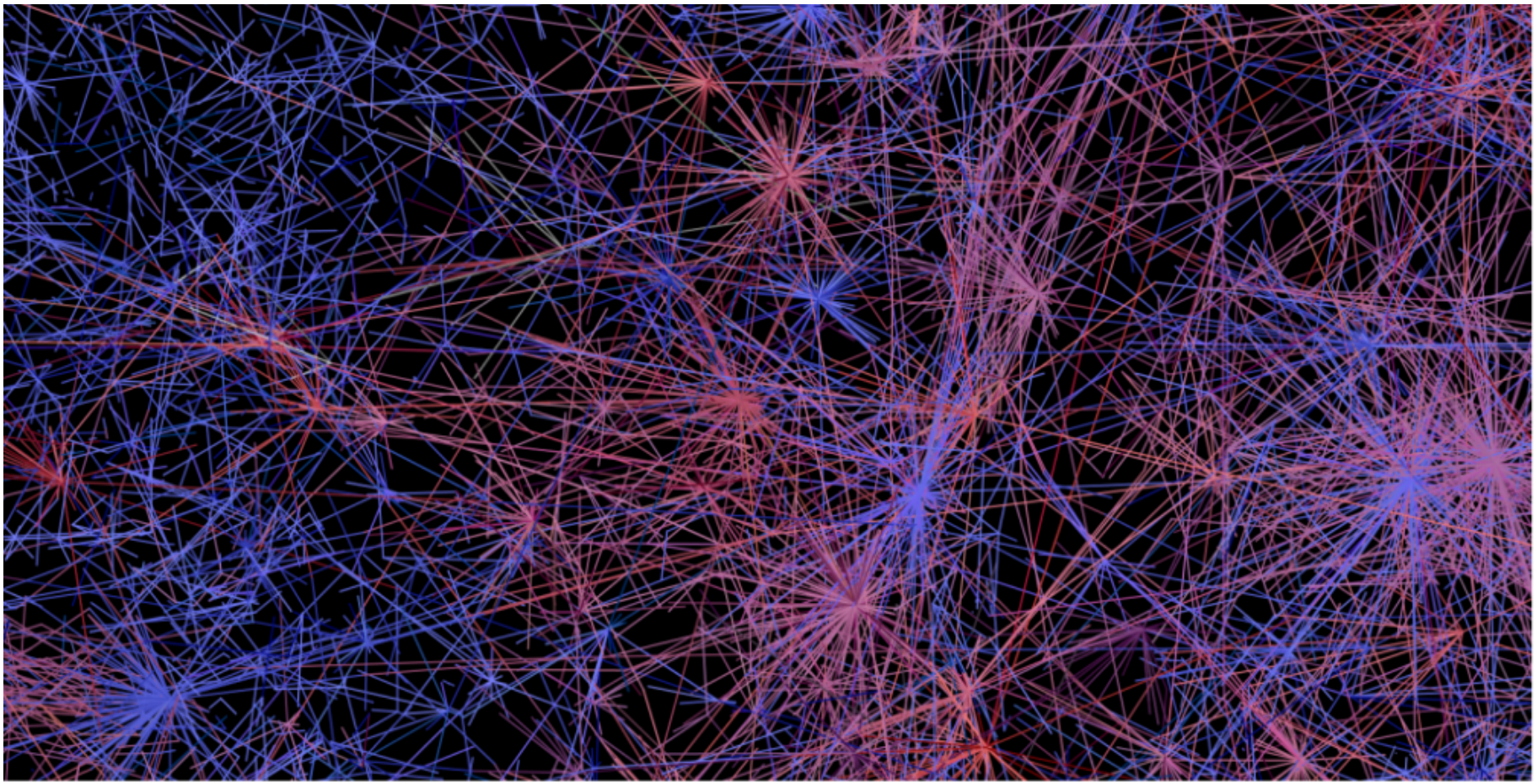
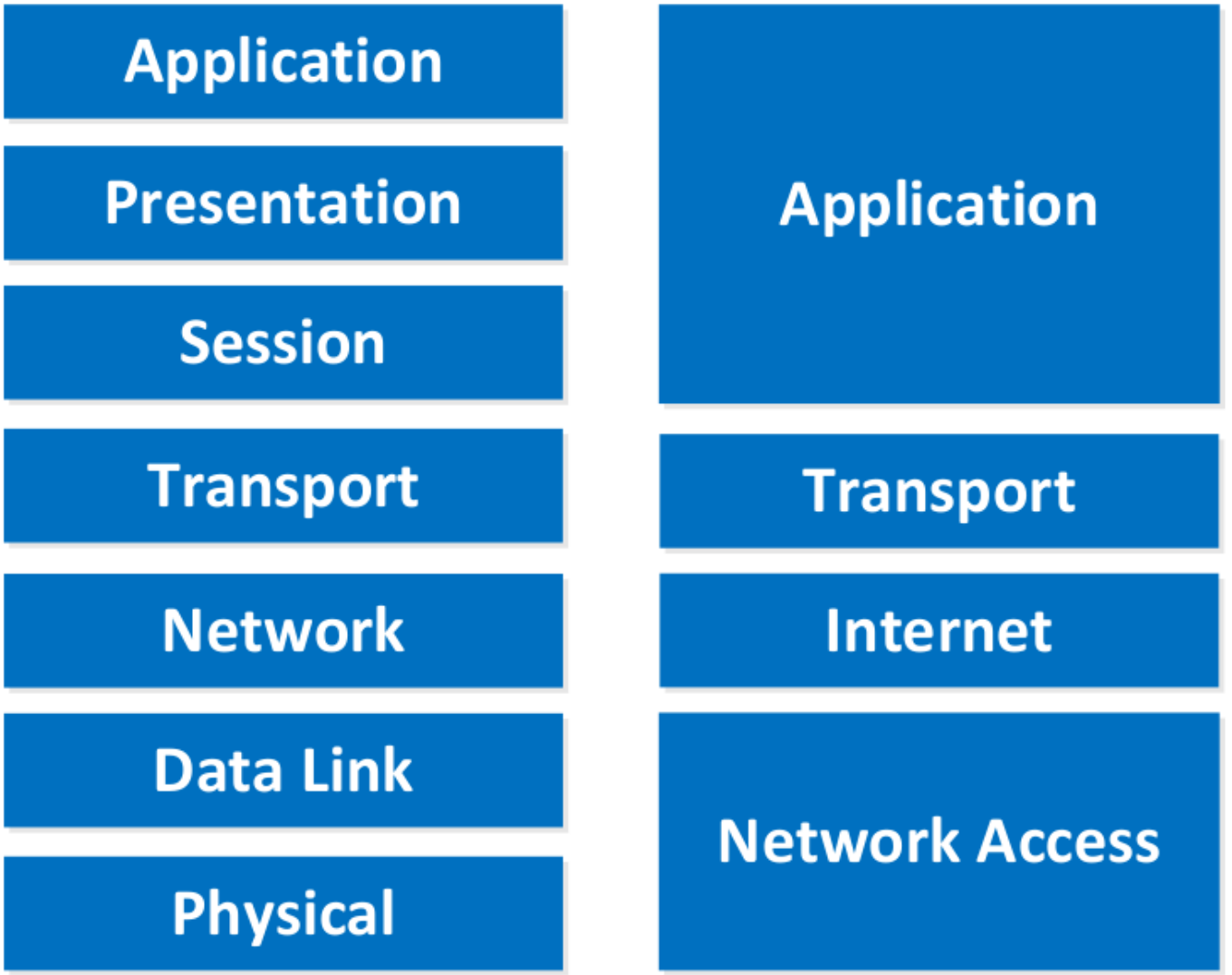
BİLGİSAYAR AĞLARI



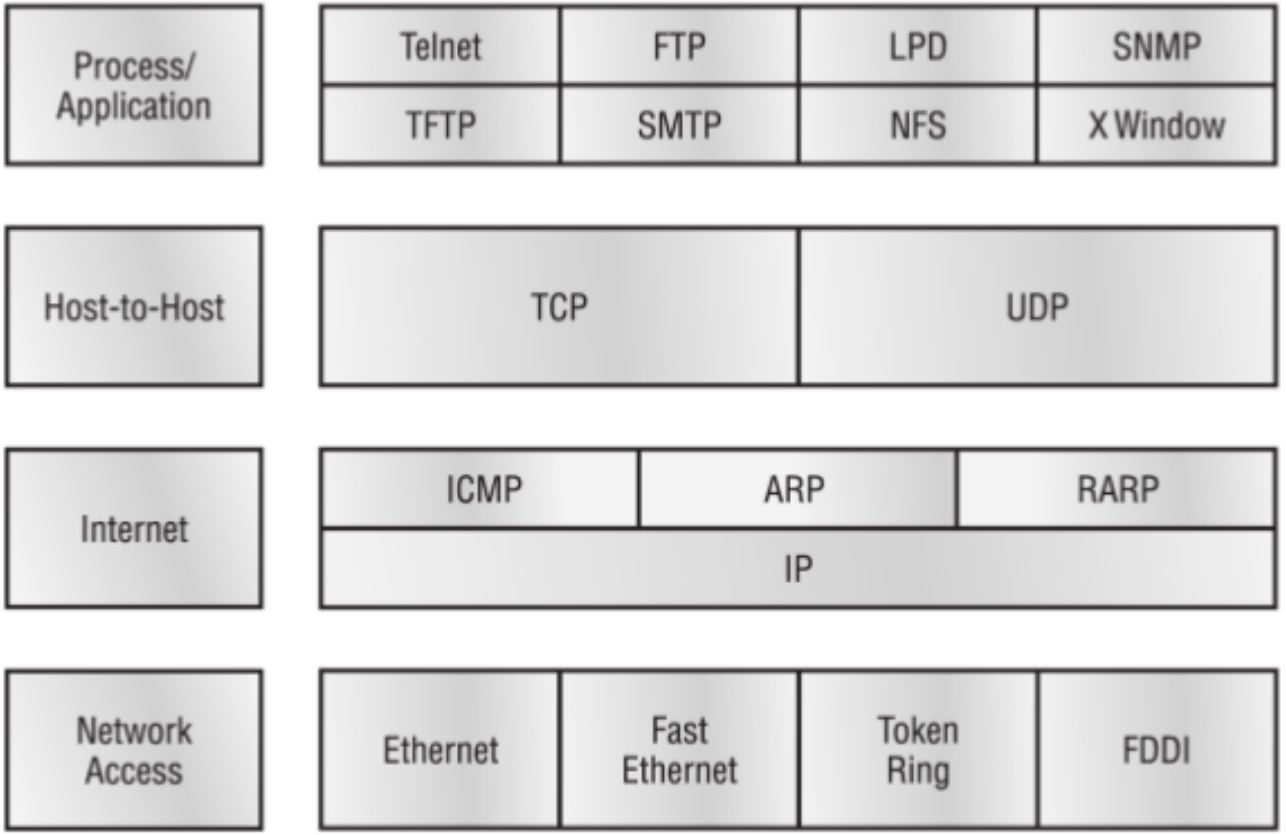
TCP/IP Mimarisi



TCP/IP Mimarisi

- OSI katmanlı mimarisiyle aynı mantıktadır. aynıdır.Sadece bazı katmanlar adları değişik ve katman sayısı azdır.
- OSI'de bulunan Physical ve Data Link Network Acceces'de denk gelir.
- OSI'de bulunan Network katmanı Internet'e denk gelir
- OSI'de bulunan Session , Presentation ve Application , Application'a denk gelir.

Katmanlar ve Protokoller



Application Katman Protokolleri

Bu katmanda veriyi göndermek isteyen uygulama ve kullandığı dosya biçimi bulunarak gönderilen verinin türüne göre farklı protokoller çalıştırılır (HTTP, SMTP, FTP, Telnet, vs.) ve programlarla Taşıma protokollerinin haberleşmesi sağlanır. Uygulama Katmanı Taşıma Katmanıyla portlar aracılığıyla haberleşir. Portlar numaralandırılmış standart uygulamalardır (HTTP:80, FTP:21, vs.) ve Taşıma Katmanında gelen paket içeriğinin türünün anlaşılmasında rol oynar.

Application Katman Protokolleri

- **FTP** (File Transfer Protocol) : Türkçe karşılığı Dosya Transfer Protokolü'dür. İsminden de anlaşılabileceği gibi internete bağlı iki bilgisayar arasında dosya transferini sağlayan bir protokol ve bu işleme hizmet eden uygulamaya verilen isimdir.**Örneğin bir web sitenize yer alması istenen dosyalar sunuculara FTP üzerinden aktarılabilir**.Alternatif olarak e-posta ya da benzer uygulamalarla göndermek için büyük olan dosyalar da, bu dosyayı size sağlanan kullanıcı adı ve şifre ile ilgili FTP sunucusuna gönderebilirsiniz. Dosya aktarımı pek çok alternatifinden farklı olarak internet bağlantısı olmadığı zaman kesilir ve bağlantı kurulduğunda kaldığı yerden devam eder.FTP protokolü ile bir başka bilgisayardan bir başka bilgisayara dosya aktarımı yapılırken, o bilgisayar ile etkileşimli-aynı anda (on-line) bağlantı kurulur ve protokol ile sağlanan bir dizi komutlar yardımıyla iki bilgisayar arasında dosya alma/gönderme işlemleri yapılır.Port numarası 21 dir

Application Katman Protokolleri

FTP'yi Ne Zaman Kullanmalıyız?

San Francisco ofisinizdeki çalışanların, e-mail ile göndermeniz gereken 50MB boyutunda bir dosyaya ihtiyaçları var. Ne yaparsınız? Birçok e-mail sunucusu, sınırlı boyut kısıtlamalarınızdan dolayı bu dosyayı göndermenize izin vermeyecektir. Sunucuda, boyut sınırı olmasa dahi, bu büyüklükte bir dosyayı San Fransisco'ya göndermek uzun bir zaman alacaktır. İşte FTP, burada hayat kurtarır!

Application Katman Protokolleri

- **TFTP** (Trivial File Transfer Protocol) : Önemsiz dosya transfer protokoludur. Aslinda **FTP** nin basit bir formatindan baska bir şey değildir. FTP deki guvenlik kisimlari cikartildiginda ortada kalan haline TFTP denir. TFTP , UDP protokolunu kullaniyor. Yukardada dedigim gibi **TFTP , FTP** nin basitlestirilmis halidir. Amaç aslinda UDP protokolunu destekleyen iki makina arasinda dosya transferi yapabilmek. Bunun haricinde baska bir seyi desteklemez **TFTP. FTP** deki gibi login olmana gerek yoktur Sadece dosya transferi yapabilirsin o kadar.Port numaras 69 dur

Application Katman Protokolleri

- HTTP (Hypertext Transfer Protocol (HTTP) (Köprü Metni Aktarım Protokolü) : Web sunucuları ile Web tarayıcılarının Internet üstünden birbirleri ile haberleşmek için kullandıkları ortak dildir. Bu protokol kullanıcı ile sunucu arasındaki mesaj alısverisinin hangi formatta olacagını belirler. Güvenilir bilgi iletimi gerektirdigi için mesajlar TCP baglantısı ile iletilir. Dolayısıyla kullanıcı ile sunucu arasında bir bilgi alısverisi olmadan önce, iki nokta arasında TCP baglantısı kurulur. 80 numaralı TCP portunu kullanır.Web siteleri büyük küçük bir çok dosyadan oluşur.Bir istek geldiğinde bu dosyaların hızlı bir şekilde aktarılması gerekir.Bu aktarma işleminde ftp yetersiz kaldığı için HTTP protokolü tanımlanmıştır.

Application Katman Protokolleri

- Telnet:Telnet protokolü ağ üzerindeki çok kullanıcılı bir makineye (sunucu-server) başka bir makina (pc) kullanarak bağlanmak ve o makine üzerinde bazı komutları çalıştırmak için kullanılır. TCP/IP protokollerinden birisidir. Telnet 23 numaralı portu kullanır. Telnet ile bağlanmak istediğimiz makine üzerinde kullanıcı adı ve şifremizin olması gerekmektedir. Veriler şifrelenmez. Bundan dolayı güvensiz bir protokoldür. Ağımızı dinlemek isteyen herhangi biri verilerimizi görebilir. Telnet protokolünü kullanmak için gereken programlar işletim sistemlerinde mevcuttur.

Application Katman Protokolleri

- SSH : Bir bilgisayarın aynı ağda bulunan bir sunucuya uzaktan bağlanmasını sağlayan bir protokoldür. TELNET protokolü de bu işi yapıyor fakat TELNET şifresiz SSH şifreli olarak çalışır. Yani bir sunucuya bağlantı yaparken kullanıcı adı ve şifreler açık metin olarak değil şifrelenmiş olarak iletilir.

Transport Katman Protokolleri

Bu katman verinin alınmasına , iletilmesine ve hata kontrolleri ile ilgilenir.Bu katmanda alınan paketlerde bir eksiklik olup olmadığı , sırasının doğruluğu ve paketlerde herhangi bir hatanın olup olmadığını kontrol eder.Bu katmanda gönderilen veya alınan paketlerin eksikliği veya hatalı olması şeklinde tekrar gönderilmesi sağlanır.

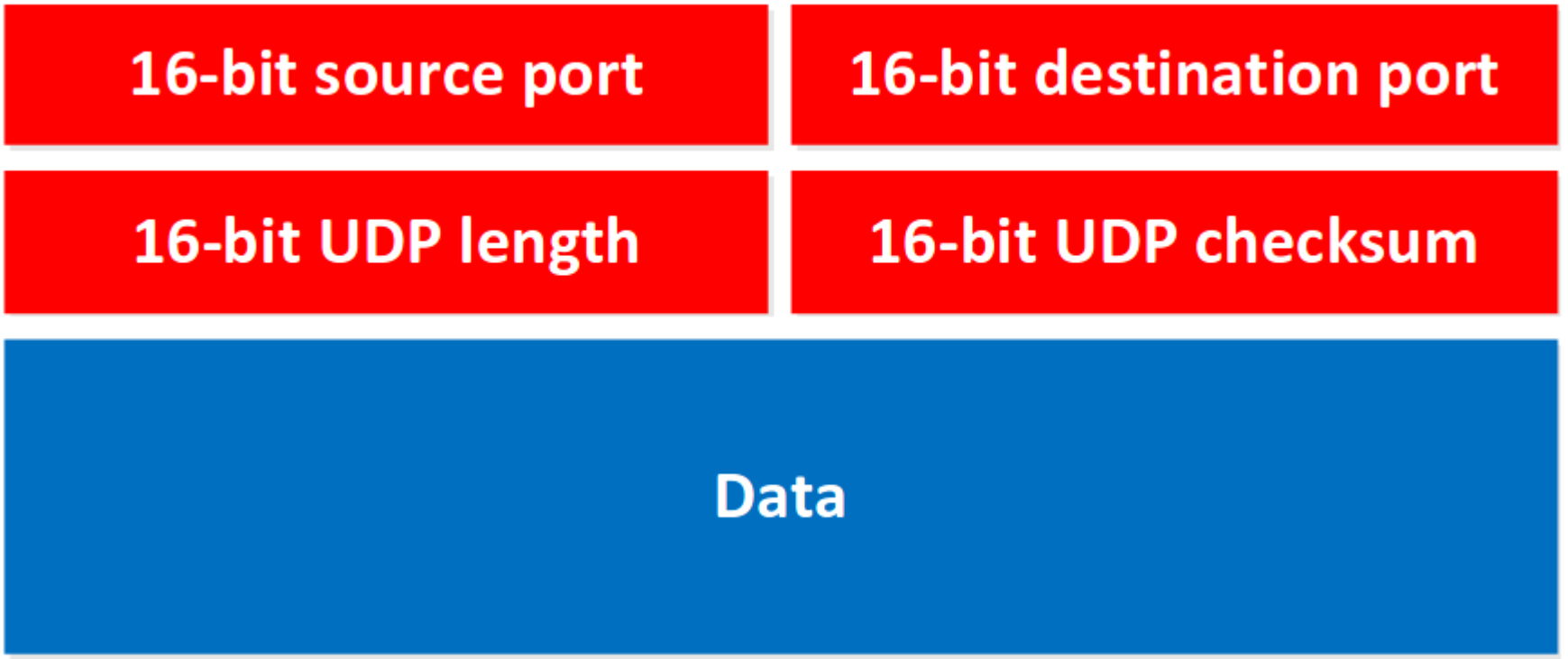
Transport Katman Protokolleri

TCP	UDP
Reliable	Unreliable
Connection-oriented	Connectionless
Segment retransmission and flow control through windowing	No windowing or retransmission
Segment sequencing	No sequencing
Acknowledge segments	No acknowledgement

UDP (User Datagram Protocol)

- Transport katmanı üzerinde çalışır
- Connectionless(oturum kurmaz) bir yapısı vardır
- Daha hızlı veri transfer etmek için kullanılabilir.
- Sanal devre oluşturmaz
- UDP, segment’leri sıraya almaz ve segment’lerin, hedefe hangi sırayla ulaşacağıyla ilgilenmez.UDP, segment’leri gönderir ve onları unutur. Takip etmez, onları kontrol etmez veya güvenli erişim onayına bile izin vermez. Bundan dolayı, güvenilmez bir protokol olarak belirtilir. Bu UDP’nin faydasız bir protokol olduğu anlamına gelmez, sadece güvenlik sorunlarını ele almaz.

UDP (User Datagram Protocol)



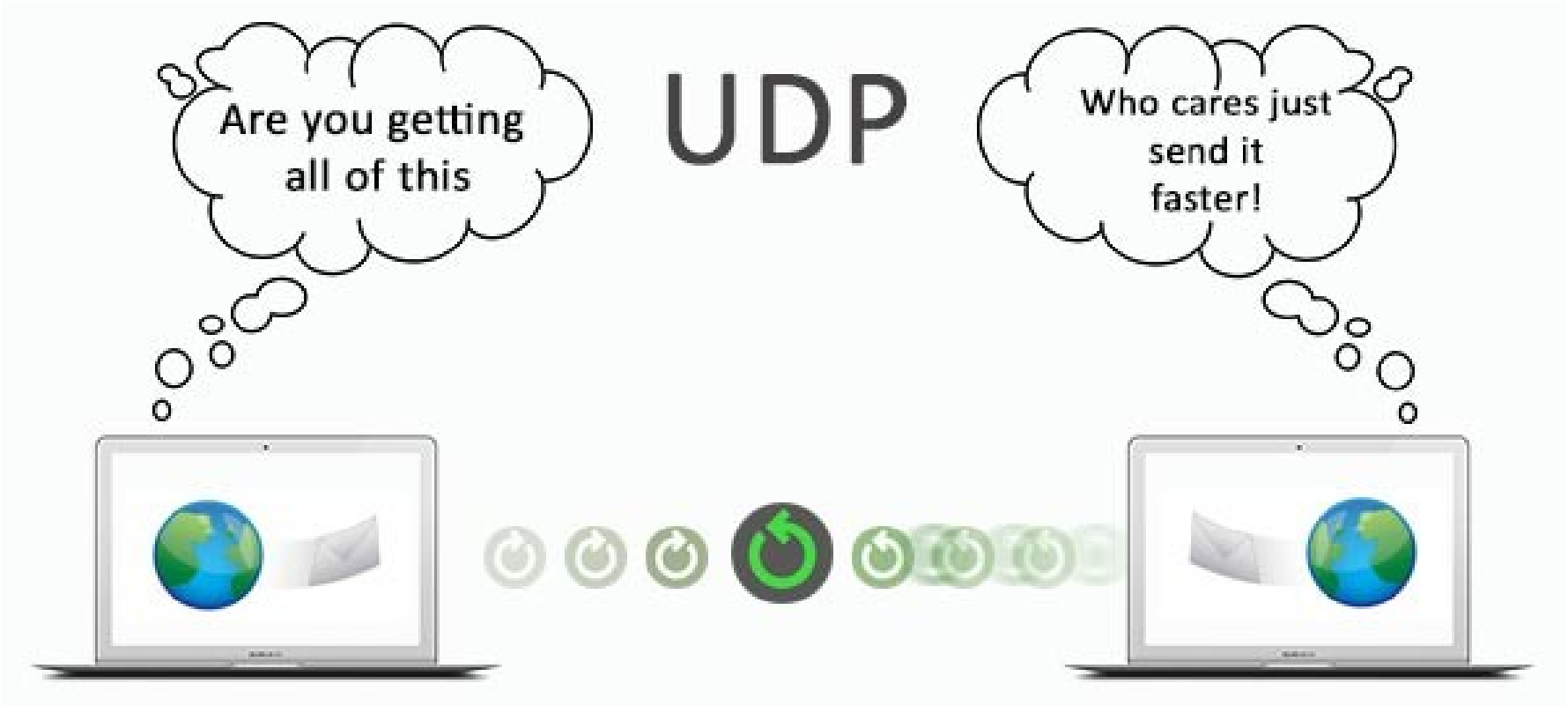
UDP (User Datagram Protocol)

- Source Port :Kaynak makine’nin hangi portundan çıktığı bilgisidir
- Destination Port:Hedef makinede’nin hangi port’undan alacağı bilgisidir.
- Length : UDP başlığı ve UDP verisinin uzunluğu
- Checksum : Hem UDP başlığı hem de UDP veri alanının sağlaması.
- Data : Üst-katman verisi

UDP (User Datagram Protocol)

```
UDP - User Datagram Protocol
Source Port:      1085
Destination Port: 5136
Length:           41
Checksum:         0x7a3c
UDP Data Area:
..Z.....00 01 5a 96 00 01 00 00 00 00 00 11 0000 00
...C..2._C._C  2e 03 00 43 02 1e 32 0a 00 0a 00 80 43 00 80
Frame Check Sequence: 0x00000000
```

UDP (User Datagram Protocol)



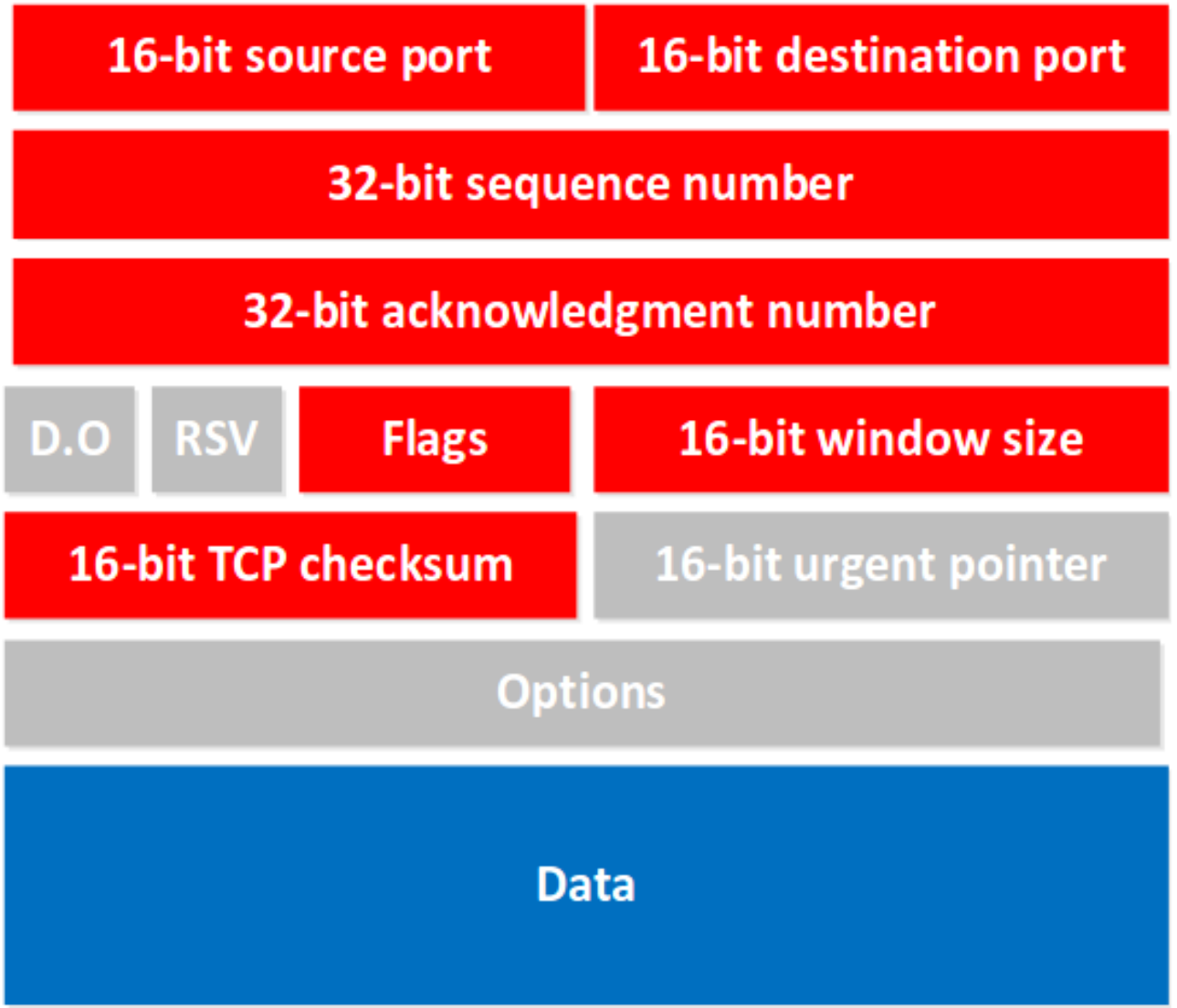
TCP (Transmission Control Protocol)

- Bir uygulamadan(Application katmanından), geniş bilgi parçalarını alır ve onları segment'lere ayırır. Her segment'i numaralar ve sıralar, böylece hedefin TCP yığını, segment'leri tasarlanan uygulama için tekrar sıraya koyar. Bu segment'lerin gönderilmesinden sonra, TCP (verici makinesinde), alıcı ucun TCP sanal devre oturumu için bir acknowledgment bekler, onaylanmayanları tekrar gönderir

TCP (Transmission Control Protocol)

- Verici makinelerin segment'leri modele göndermeye başlamasından önce, göndericinin TCP yığını, bir bağlantı oluşturmak için hedefin TCP yığını ile iletişime geçer. Oluşturulan, sanal bir devre olarak bilinir. Bu iletişim tipi, connection-oriented'dir. Başlangıç anlaşması esnasında, iki TCP katmanı, alıcının TCP'sinin, tekrar bir acknowledgment göndermesinden önce gönderilecek bilgi miktarı konusunda anlaşır. Önceden her konuda anlaşarak, güvenli iletişim olması için bir yol oluşturulur.

TCP (Transmission Control Protocol)



TCP (Transmission Control Protocol)

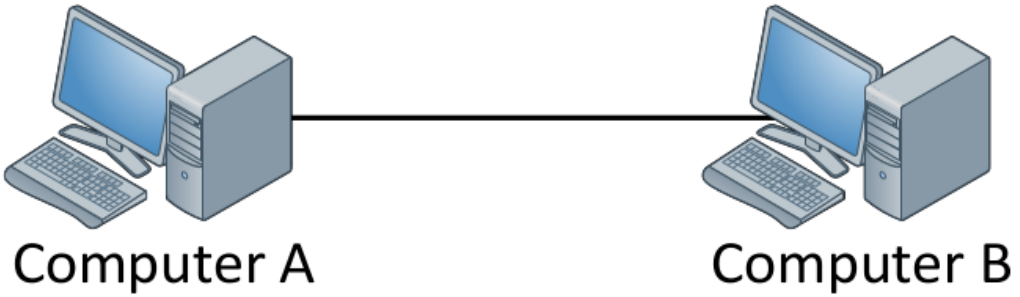
- Source Port :Kaynak makine’nin hangi portundan çıktığı bilgisidir
- Destination Port:Hedef makinede’nin hangi port’undan alacağı bilgisidir.
- Sequence number: Veriyi doğru sıraya geri koyan ya da kayıp veya bozuk veriyi tekrar ileten
- Acknowledgment number: Bir sonraki olması beklenen TCP okteti.
- Window: Göndericinin kabul edeceği, oktetlerdeki window boyutu.
- Checksum: TCP, alt katmanlara güvenmediğinden, CRC (cyclic redundancy check), her şeyi denetler. CRC, başlık ve veri alanlarını kontrol eder
- Flags : TCP’nin farklı tip mesajları göndermesi için belirlenen bilgiler içerir
- Data: Transport katmanındaki TCP protokolüne gönderilir, üst katman başlıklarını içerir.

TCP (Transmission Control Protocol)

```
TCP - Transport Control Protocol
 Source Port:      5973
 Destination Port: 23
 Sequence Number:  1456389907
 Ack Number:       1242056456
 Offset:           5
  Reserved:        %000000
  Code:            %011000
         Ack is valid
         Push Request
  Window:          61320
  Checksum:        0x61a6
  Urgent Pointer:  0
  No TCP Options
  TCP Data Area:
  vL.5.+.5.+.5.+.5  76 4c 19 35 11 2b 19 35 11 2b 19 35 11
   2b 19 35 +. 11 2b 19
  Frame Check Sequence: 0x0d00000f
```

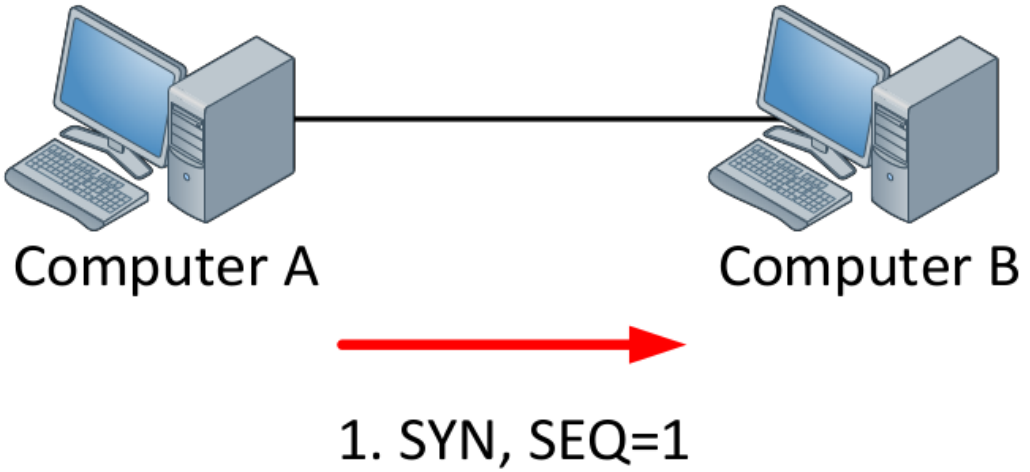
TCP (Transmission Control Protocol)

Now let's see what TCP can offer us. First of all since TCP is a reliable protocol it will "setup" a connection before we start sending any data. This connection is called the "**3 way handshake**".



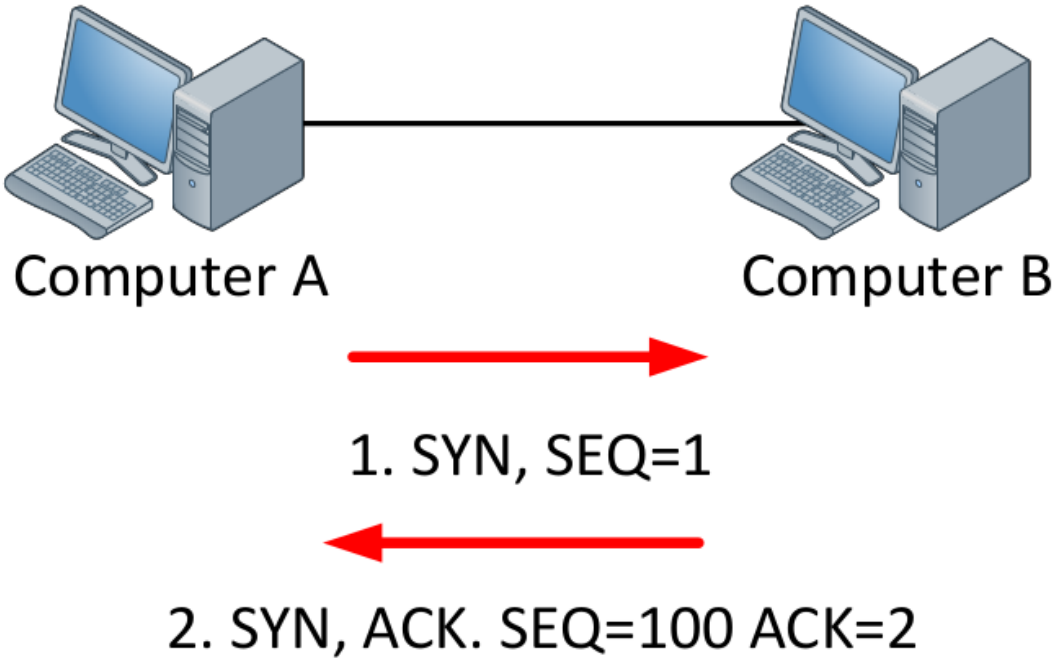
TCP (Transmission Control Protocol)

Computer A wants to send data to computer B in a reliable way, so we are going to use TCP to accomplish this. First we will setup the connection by using a 3-way handshake, let me walk you through the process:



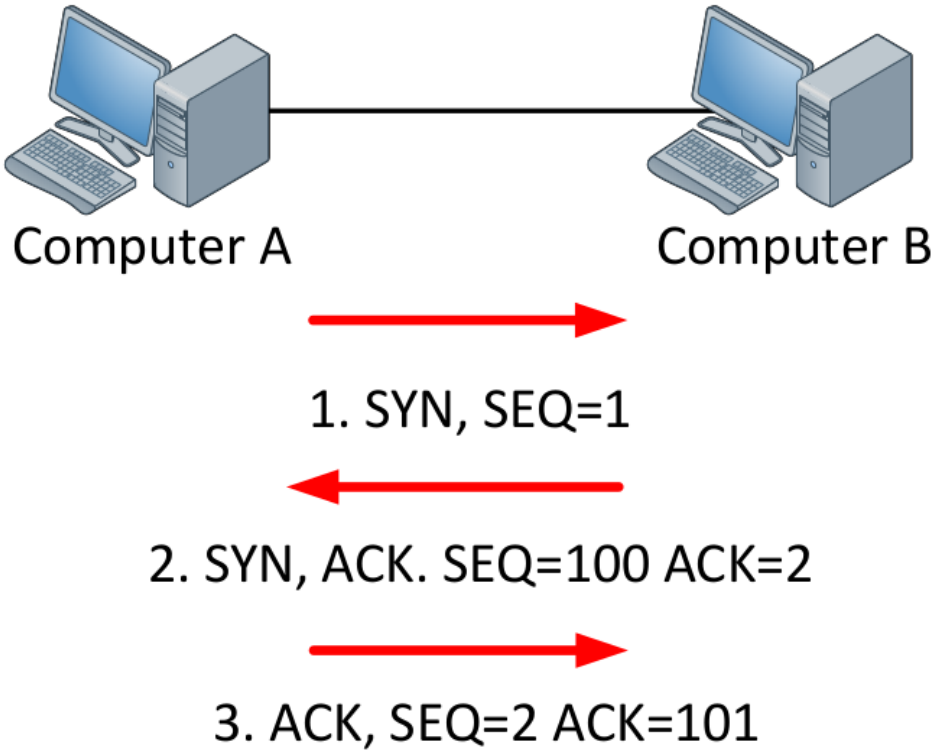
TCP (Transmission Control Protocol)

First our computer A will send a **TCP SYN**, telling computer B that it wants to setup a connection. There's also a sequence number and to keep things simple I picked number 1.



TCP (Transmission Control Protocol)

ACK=2 means that it acknowledges that it has received the TCP SYN from computer A which had sequence number 1 and that it is ready for the next message with sequence number 2.



TCP (Transmission Control Protocol)

- Computer A sends a **TCP SYN**. (I want to talk to you)
- Computer B sends a **TCP SYN,ACK**. (I accept that you want to talk to me, and I want to talk to you as well)
- Computer A sends a **TCP ACK**. (I accept that you want to talk to me)

t me show you an example in Wireshark what this looks like on a real network:

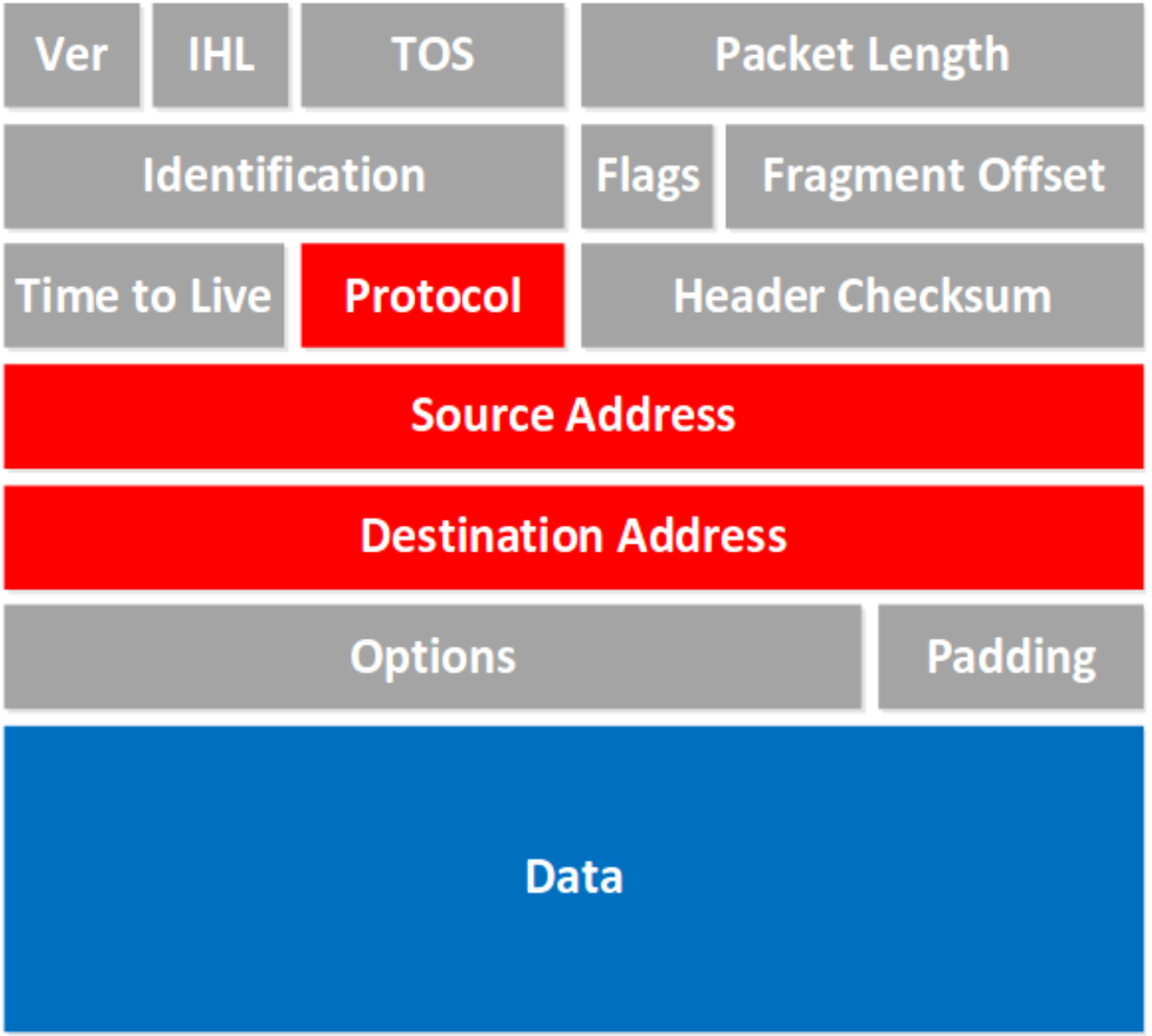
Network Katman Protokolleri

- Network katmanı:Network :Bu katman bağlantıyı ve yol seçimini belirler.Ayrıca bu katman mantıksal adreslerin(MAC) fiziksel adreslere (IPv4 , IPv6) adreslerine çevirilmesini sağlar.Bu katmanda gönderilen verinin hedefe ulaşıncaya kadar en iyi yolları seçerek gider.

IP (Internet Protocol)

- IP (Internet Protocol) : Gönderilecek olan paketlerin destination ip adresine bakarak nereye gideceklerini tanımlar.
- IP, her paketin adresine bakar. Sonra, bir routing tablosu kullanarak en iyi yolu seçip bir paketin nereye gönderileceğine karar verir.
- Ağ üzerindeki her ağ aygıtını benzersiz şekilde tanımlamak için bir IP adresine ihtiyacımız var. (Ev telefonlarının benzersiz numaraları gibi)

IP (Internet Protocol)



IP (Internet Protocol)

- Versiyon: IP versiyon numarası.
- Flags: Parçalanmanın olup olmadığını belirtir.
- Fragment offset: Şayet, paket bir frame'e koymak için çok büyükse parçalama ve tekrar bir araya getirme sağlar. Aynı zamanda, internette farklı MTU'lara (maximum transmission unit) izin verir
- Time to Live: Time to Live, orijinal olarak üretildiğinde, bir pakete ayarlanır. TTL süresinin dolmasından önce, istediği yere gitmezse, paket iptal olur. Bu, IP paketlerinin ağda sürekli dolaşımını durdurur
- Protocol: Üst katman protokolün portudur (TCP, port 6'dır veya UDP, port 17'dir [hex]). Ayrıca ARP ve ICMP gibi, Network katmanı protokollerini destekler.
- Source IP address: Gönderen istasyonun 32-bit IP adresi.
- Destination IP address: Bu paketin hedeflendiği istasyonun 32-bit IP adresi.
- Data: IP seçeneğinden sonraki alan, üst-katman verisi olacaktır.

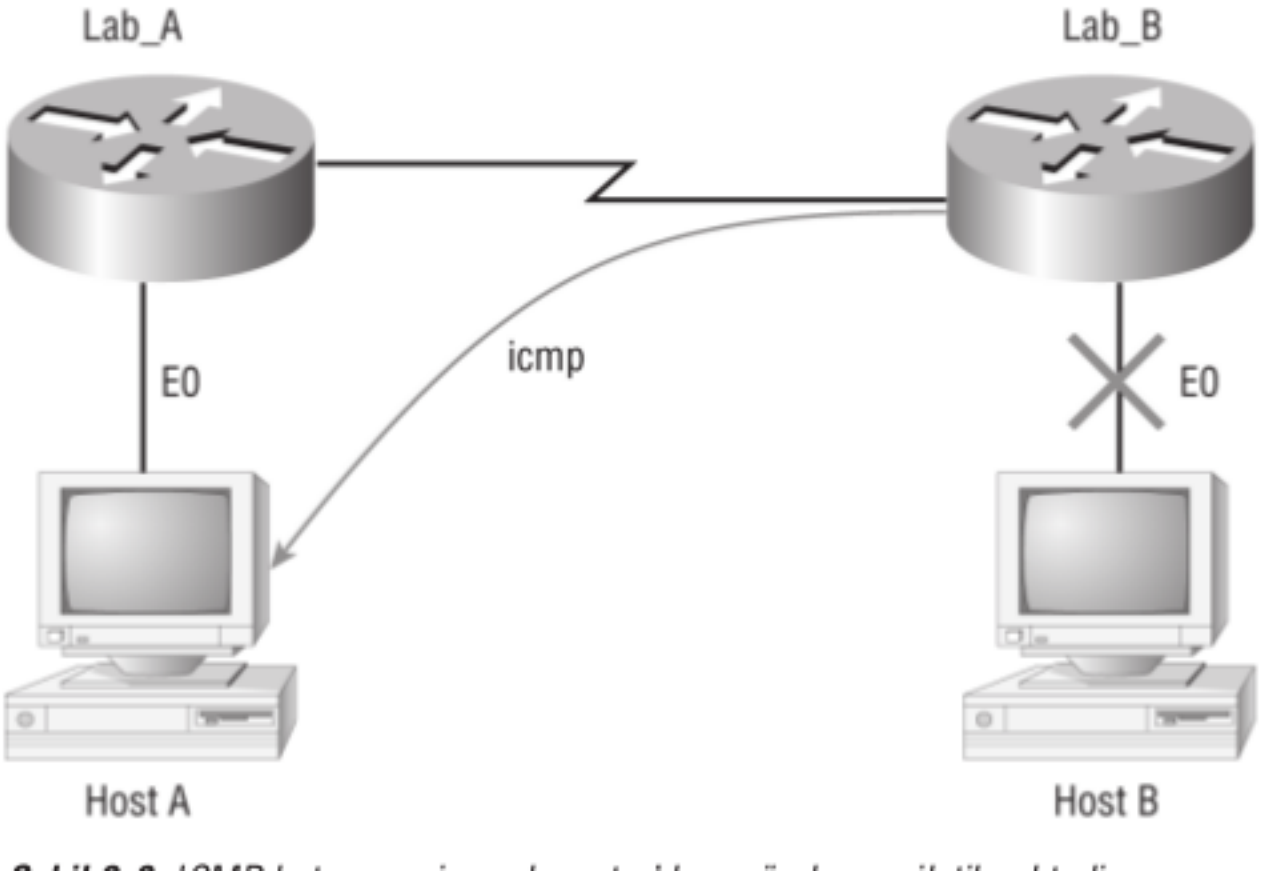
IP (Internet Protocol)

```
IP Header - Internet Protocol Datagram
 Version:              4
 Header Length:        5
 Precedence:           0
 Type of Service:      %000
 Unused:               %00
 Total Length:         187
 Identifier:           22486
 Fragmentation Flags: %010 Do Not Fragment
 Fragment Offset:      0
 Time To Live:         60
 IP Type:              0x06 TCP
 Header Checksum:      0xd031
 Source IP Address:    10.7.1.30
 Dest. IP Address:     10.7.1.10
 No Internet Datagram Options
```

ICMP(Internet Control Message Protocol)

- Internet Control Message Protocol (ICMP), Network katmanında çalışır ve birçok farklı servis için IP tarafından kullanılır.
- Kullanıcı makinelerine, ağ problemleri hakkında bilgi sağlar.
- *IP datagram'larında enkapsüle edilirler.
- Destination Unreachable (Hedef erişilemez): Şayet bir router, artık bir IP datagram gönde-remezse, göndericiye, durumunu belirten bir mesaj göndermek için ICMP'yi kullanır.
- ICMP request : İstek atmak için gönderilir
- ECMP reply : İstekleri cevaplamak için gönderilir

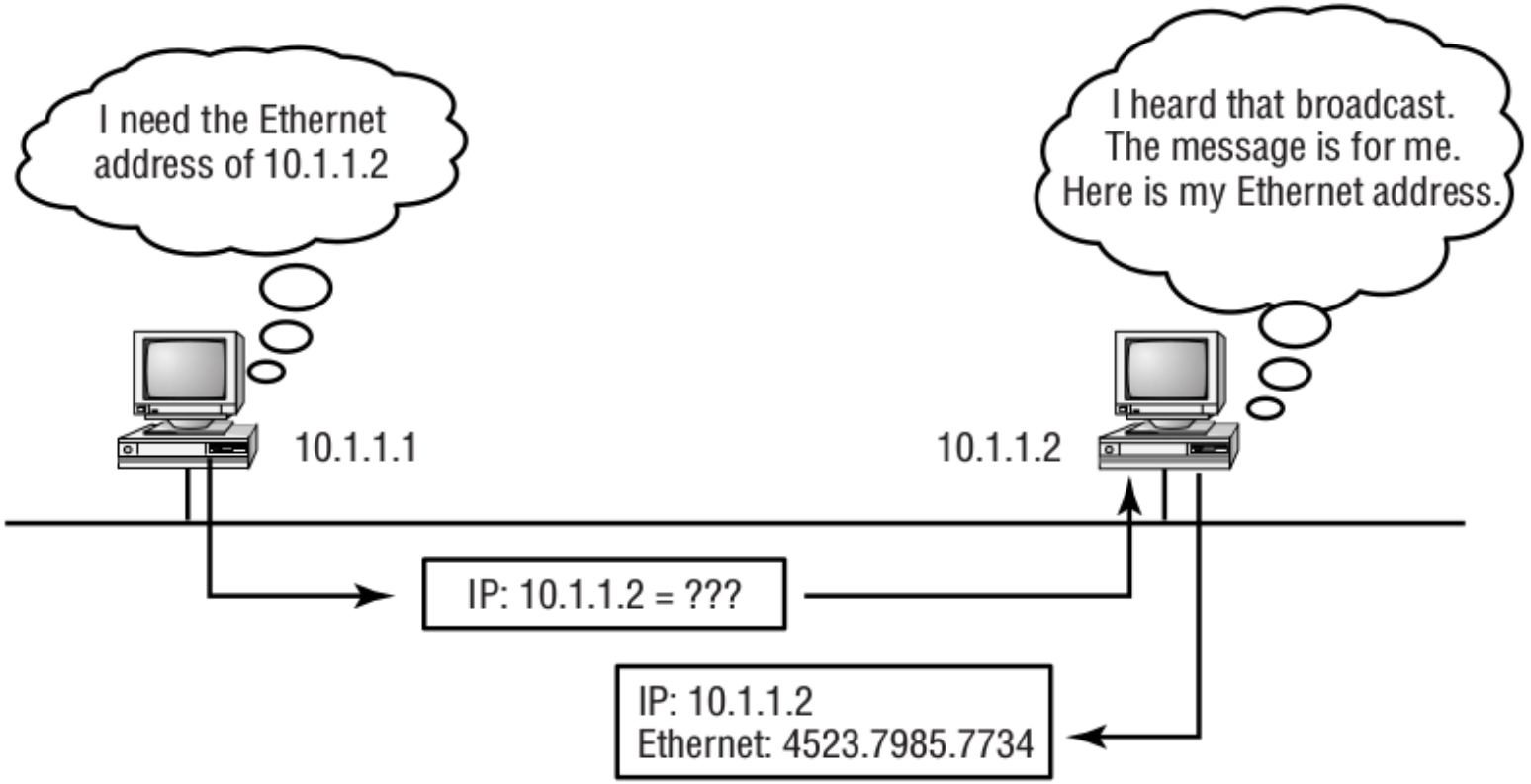
ICMP(Internet Control Message Protocol)



ARP (Address Resolution Protocol)

- Bilinen bir host'un IP adresini , donanım adresine çevirir. (IP → MAC)
- Eğer hedef makinanın IP adresi ARP cache'de yoksa donanım adresinin bulunması için ARP kullanılır.
- IP'nin dedektifi gibi, ARP, belirli bir IP adresi ile sorduğu makinenin donanım adresini isteyen bir broadcast göndererek yerel ağı sorgular. Aslında ARP, yazılım (IP) adresini bir donanım adresine (örneğin hedef makinenin Ethernet bord adresine) çevirir ve bundan, adres için broadcast göndererek LAN'daki yeri hakkında sonuç çıkarır

ARP (Address Resolution Protocol)



ARP (Address Resolution Protocol)

```
Flags:        0x00
  Status:       0x00
  Packet Length:64
  Timestamp:    09:17:29.574000 01/04/2000
Ethernet Header
  Destination:  FF:FF:FF:FF:FF:FF  Ethernet Broadcast
  Source:       00:A0:24:48:60:A5
  Protocol Type:0x0806  IP ARP
ARP - Address Resolution Protocol
  Hardware:                  1  Ethernet (10Mb)
  Protocol:                  0x0800  IP
  Hardware Address Length:   6
  Protocol Address Length:   4
  Operation:                 1  ARP Request
  Sender Hardware Address:   00:A0:24:48:60:A5
  Sender Internet Address:   172.16.10.3
  Target Hardware Address:   00:00:00:00:00:00  (ignored)
  Target Internet Address:   172.16.10.10
Extra bytes (Padding):
  ..............  0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A
0A 0A 0A 0A 0A
Frame Check Sequence:  0x00000000
```

RARP(Reverse Address Resolution Protocol)

- RARP protokolü bir sunucunun donanım adresleri verilip protokol adreslerini çözümler.

Data Link Katman Protokolleri

- Gönderilecek verinin ağ ortamında nasil iletileceğini, fiziksel adreslemeyi ve ağ topolojisini tanımlar. Fiziksel kabloda çarpışma olmadan veri iletimini sağlamak (CSMA/CD) ya da verilerin bu medyaya nasıl konulacağı belirlemek ve yine verinin hatalara karşı kontrolünü yapmak(CRC), ağ üzerindeki diğer pc'lerin kimlik doğrulamalarını yapmak, anlık iletişimin kimin tarafından yapıldığını tespit etmek bu katmanın görevidir.
- 3. katmanda (ağ katmanı) paketlere dönüştürülen veri (data), bu katmanda artık fiziksel ortama aktarılmadan önce son kez işlem görerek çerçeve (frame) yapılara dönüştürülür. Çerçeveler verileri belli bir kontrol içinde göndermeyi sağlayan yapılardır. Veri bağlantı katmanı üzerinden iletimi yapılan her paket, kaynak (source) ve hedef (destination) adreslerini içerir, yani her paket başlangıcı ve bitişi belli olacak şekilde özel bitlerle işaretlenir.

LLC(Logical Link Control)

- Bozulmuş olarak giden paketlerin tekrar gönderilmesini sağlamak temel görevlerindendir.
- Alıcının işleyebileğinden fazla veri paketi gönderilerek boğulmasının engellenmesinden de LLC sorumludur.
- İlgili Protokole özel mantıksal portlar oluşturur (Service AccessPoints, SAP). Verinin kapsüllenmesi sırasında ağ katmanlarından gelen veriye hedef ve kaynak protokol bilgilerini ekler ve tekrar paketlediği veriyi bir alt katman olan Ortam Giriş Kontrol (Media Access Control-MAC) katmanına aktarır. Böylece kaynak makinada ve hedef makinada aynı protokoller iletişime geçebilir
- Bu günlerde LLC'nın yerini artık TCP dolduruyor(Hatalı veri kontrol , Flow kontrol).

MAC(Media Access Control)

- Bir MAC adresi, üretici tarafından bir ağ donanımına (kablosuz kart veya ethernet kartı gibi) verilen benzersiz tanımlayıcıdır. MAC, Ortam Erişim Kontrolü anlamına gelir ve her tanımlayıcı belirli bir aygıta özeldir.
- MAC adresi altı karakterden oluşan iki karakterden oluşur ve her biri iki nokta üst üste ile ayrılmıştır. 00: 1B: 44: 11: 3A: B7 bir MAC adresinin bir örneğidir